Extensions to the GrETEL Treebank Query Application

Jan Odijk
Utrecht University / Utrecht
j.odijk@uu.nl

Martijn van der Klis
Utrecht University / Utrecht
m.h.vanderklis@uu.nl

Sheean Spoel
Utrecht University / Utrecht
s.j.j.spoel@uu.nl

Abstract

In this paper we describe the extensions we made to an existing treebank query application (GrETEL). These extensions address user needs expressed by multiple linguistic researchers and include (1) facilities for uploading one's own data and metadata in GrETEL; (2) conversion and cleaning modules for uploading data in the CHAT format; (3) new facilities for analysing the results of the treebank queries in terms of data, metadata and combinations of them. These extensions have been made available in a new version (Version 4) of GrETEL.

1 Introduction

In this paper we describe the extensions we made to an existing treebank query application (GrETEL) in the context of the AnnCor project, in which we are (inter alia) developing a treebank for the Dutch CHILDES corpora (MacWhinney, 2000).[1] The AnnCor treebank and the treebank query application are being developed in the Utrecht University AnnCor project, which we describe in section 2.[2] We briefly describe the treebank in section 3. We describe the extensions of the treebank query application GrETEL in section 4. We illustrate the extensions by means of an example in section 5. In section 6 we discuss related work, and we end with conclusions and plans and suggestions for future work in section 7.

This paper (1) presents facilities for uploading one's own data and metadata in GrETEL; (2) describes the conversion and cleaning modules for uploading data in the CHAT format; and (3) presents new facilities for analysing the results of the treebank queries in terms of data, metadata and combinations of them.

2 The AnnCor Project

The AnnCor project is an Utrecht University internal research infrastructure project that aims to create linguistically annotated corpora for the Dutch language and to enhance and extend an existing treebank query application in order to query the annotated corpora. Various types of corpora are being annotated, and various types of annotations are being added. The corpora include learner corpora, news corpora, narrative corpora, and language acquisition corpora. Annotations include annotations for learners' errors and their corrections, discourse annotations, and full syntactic structures. In this paper we focus on the application for querying the corpora, in particular treebanks (i.e. text corpora in which each utterance is assigned a syntactic structure) and analysing the search results.

Sagae et al. (2007) state for CHILDES corpora that 'linguistic annotation of the corpora provides researchers with better means for exploring the development of grammatical constructions and their usage'. The research described in (Odijk, 2015, 2016a) illustrates this for the study of the acquisition of particular syntactic modification phenomena using the Dutch CHILDES corpora. It is clear from these papers that such research cannot be done properly and efficiently without treebanks for these corpora. The AnnCor project aims to create exactly such treebanks, which, together with a query application, will

[1] The Dutch CHILDES Corpora are accessible via http://childes.talkbank.org/access/Dutch/.

[2] This paper contains many hyperlinks hidden under terms and acronyms. The presence of a hyperlink is visible in digital versions of the paper but may be badly visible or invisible in printed versions of the paper.

Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories (TLT16), pages 46–55,
Prague, Czech Republic, January 23–24, 2018.

Keywords: treebank, querying, GrETEL, CHILDES

become an integrated part of the Dutch part of the CLARIN research infrastructure (Odijk, 2016b; Odijk and van Hessen, to appear 2017). These treebanks and the associated search and analysis applications can then contribute to an acceleration of language acquisition research and to a larger empirical basis for testing theories or hypotheses.

3 The AnnCor CHILDES Treebank

The AnnCor CHILDES Treebank is created with the help of the Alpino parser (Bouma et al., 2001), which automatically assigns a syntactic structure to each utterance in the corpus. Since Alpino has been developed for written adult language such as newspapers, it is not surprising that it creates many wrong parses when applied to the CHILDES corpora.[3] The problem is twofold: CHILDES contains transcriptions of spoken utterances from dialogues, and many of them are uttered by children that are still in the process of acquiring the language. In the AnnCor project we create a manually verified subcorpus, sampled in a representative manner. In addition, we manually verified and, if needed, corrected parse trees for which it was very likely that they contain errors, as determined on the basis of a variety of heuristics for identifying potential errors. For more details about this manually verified subcorpus, we refer the reader to (Odijk et al., 2017).

3.1 Cleaning

CHILDES corpora are represented in the CHAT format (MacWhinney, 2015). Utterances in a CHAT file are enriched with all kinds of annotations. Many of these annotations are in-line annotations. Some examples are given in (1):[4]

(1) Example in-line annotations in CHILDES CHAT files:

a. < ik wi > [//] ik wil xxx bekertje doen.
< I wan > [//] I want xxx cup-DIM do
'I want to do the little cup'

b. < doe maar even > [/] doe maar even op tafel.
< put PRT PRT > [/] put PRT PRT on table
'Just put on the table'

c. knor knor [=! pig sound] , ik heb honger.
oink oink [=! pig sound] , I have hunger
'Oink oink, I am hungry'

These examples illustrate annotations for retracing ([//]) and repetition ([/]), both with scope over the preceding part between angled brackets, for unintelligible material (xxx) and for paralinguistic material ([=! ...]).

The Alpino parser cannot deal with these annotations. A cleaning programme has been developed to remove the annotations and send a cleaned utterance to the Alpino parser.

The cleaned variants of the utterances in (1) are:

(2) Example cleaned utterances:

a. ik wil xxx bekertje doen.

b. doe maar even op tafel.

c. knor knor , ik heb honger.

The cleaning program is available on GitHub[5] and has been integrated in the GrETEL upgrade described in section 4.

[3] Though even a fully automatically parsed treebank can be fruitfully used in linguistic research, as illustrated by (Odijk, 2015).

[4] The sources are indicated by the session name (e.g. Sarah35) followed by the utterance number (e.g. 224), starting counting at 1. The examples here are the utterances Sarah35.015, Sarah35.023 and Sarah35.224 from the Van Kampen corpus.

[5] https://github.com/JanOdijk/chamd.

3.2 Annotation Conventions

The utterances used by the children contain many phenomena that are considered ill-formed in the adult language. In addition, as in any annotated corpus, many phenomena can be analysed in multiple ways, none of which can be considered better than any other on purely linguistic grounds. It is important to analyse each construction in a consistent and uniform manner, so that it can be easily automatically identified and distinguished from other constructions in a treebank query application when the data are used in research. For this reason, it is important to develop and adhere to annotation conventions and guidelines.

We illustrate this with some examples of phenomena that are not part of the adult language. The following examples appear to contain a finite verb form (*lees* and *kocht*, respectively) where a participle is expected:[6]

(3) a. ik heb niet lees
I have not read-PRES
'I have not read'

b. Ik heb bolletjes kocht
I have roll-DIM-PL buy-PAST
'I have bought little rolls'

It is not a priori clear how such examples should be analysed: the child might produce forms that do not conform to the adult language due to syntactic reasons, morphological reasons or phonological reasons. One can decide among them only after an intensive investigation of the phenomena. In constructing the treebank we do not take a stand as to how such examples should be analysed, but we do treat each of them in a uniform way, so that each can be easily and automatically identified by researchers using a treebank query application. The examples in (3) are analysed in the treebank as participial verbal complements (*vc/ppart*) that contain a finite verb.

For more examples and how they are dealt with, we refer to (Odijk et al., 2017).

4 Treebank Querying

For querying the treebanks we started from the existing treebank query application *GrETEL*, which was developed in Leuven (Augustinus et al., 2012). This application comes in three versions,[7] and we started from version 3.[8] We extended this treebank search application with functionality that was requested by many linguists: they want to be able to upload their own data with metadata, in formats that they actually use (in the context of language acquisition and related fields the most frequently used format is CHAT), and not only get a list of sentences as a result of their queries but facilities for analysing the query results in terms of the relevant parts of the structures in combination with metadata. Initial versions of these extensions have been incorporated in GrETEL Version 4, and are being further refined.[9]

The existing treebank search application GrETEL allows researchers to search in Dutch treebanks and to perform a limited analysis of the search results. GrETEL has a very user-friendly example-based interface, but also allows queries in the XML query language XPath.

The example-based search interface enables one to query the treebank by providing an example sentence that illustrates the construction one is interested in, plus some information on which aspects of this sentence are crucial for the construction. The system parses the example sentence (using the same parser as the one used to create the treebank) and enables the user to select the substructure of this parse relevant for the construction.

In GrETEL 4, the corpus upload functionality was added as a separate application and allows users to upload an archived collection (zip file) of text files. The collection is subdivided in multiple components (on the basis of the folder structure). The software will tokenise and parse these files using the Alpino

[6] Utterances Laura09.527 and Laura13.042 from the VanKampen Corpus.

[7] See http://nederbooms.ccl.kuleuven.be/eng/gretel and references there for versions 1 and 2.

[8] GrETEL Version 3 can be found here: http://gretel.ccl.kuleuven.be/gretel3/index.php.

[9] GrETEL Version 4 is currently still under development but can already be used here: http://gretel.hum.uu.nl/gretel4/.

dependency parser (Bouma et al., 2001), and import them into the XML database BaseX (Grün, 2010) for querying with GrETEL. Users can specify their corpus as private (only searchable for them) or publicly available. Figure 1 shows a screenshot of the upload interface.

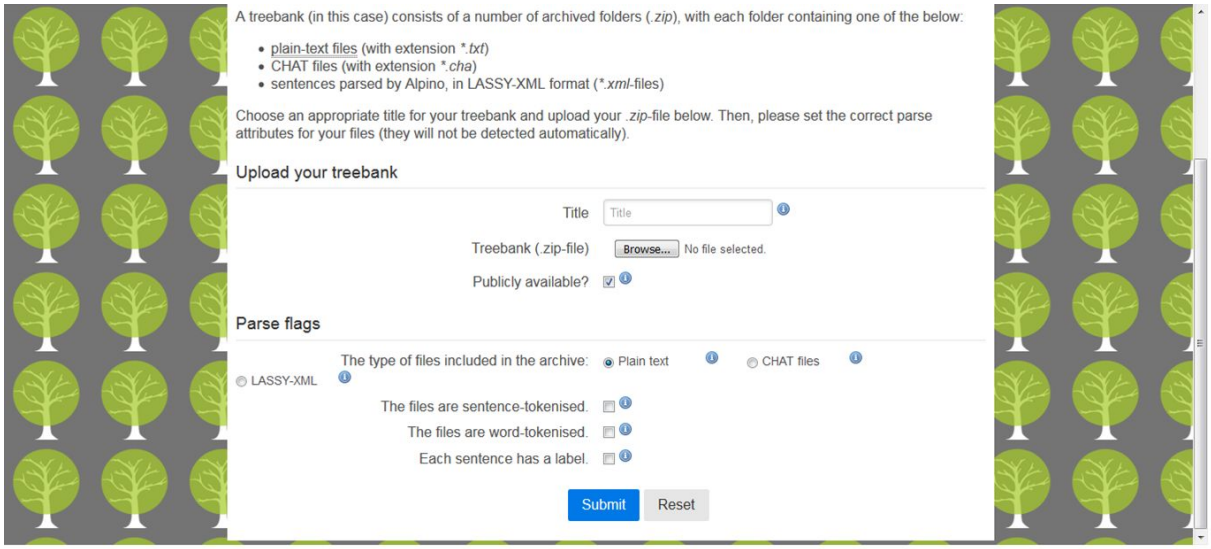


Figure 1: Screenshot of the GrETEL corpus upload page.

An interface is available to the researcher for managing the uploaded corpora. It offers buttons for viewing detailed information on the uploaded corpora, for viewing the uploading logfile, for making the corpora public, for downloading the treebank and for deleting it. A screenshot of this interface is given in Figure 2.

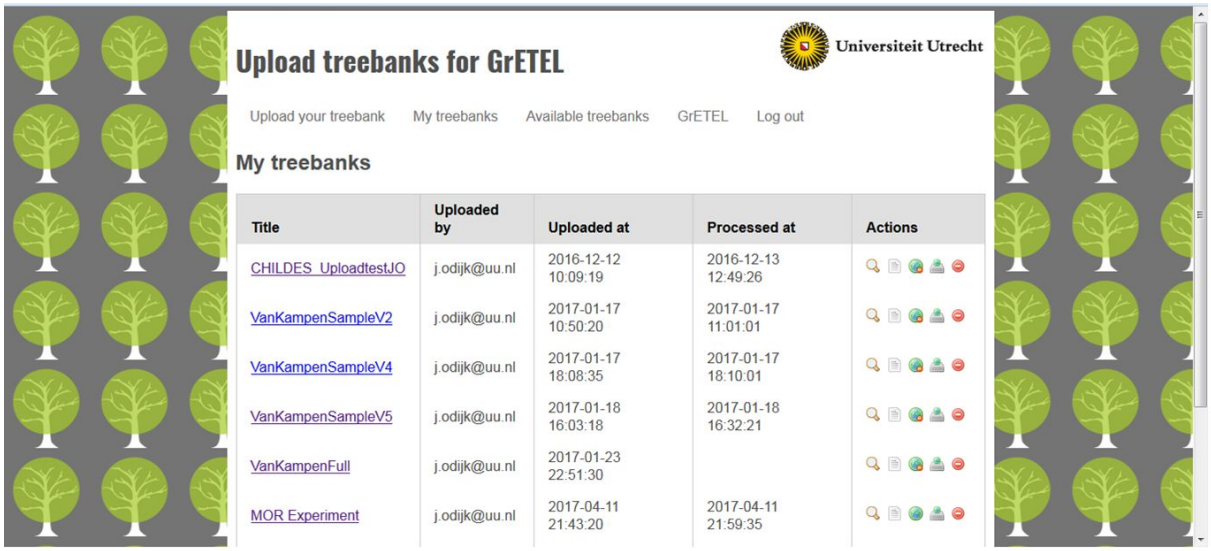


Figure 2: Screenshot of the GrETEL corpus managing page ('My treebanks').

The corpus details page (see Figure 3 for a screenshot) contains information about the components the treebank consists of and about the size of each component of the treebank (# sentences, # word occurrences). It also offers the user the option to select which metadata elements will occur in the analysis component and which user interface option is used for selecting values for a specific metadata element, e.g. to use a range filter instead of checkboxes for numeric metadata.

In GrETEL 4, one can upload a treebank parsed with Alpino (with XML files in accordance with the

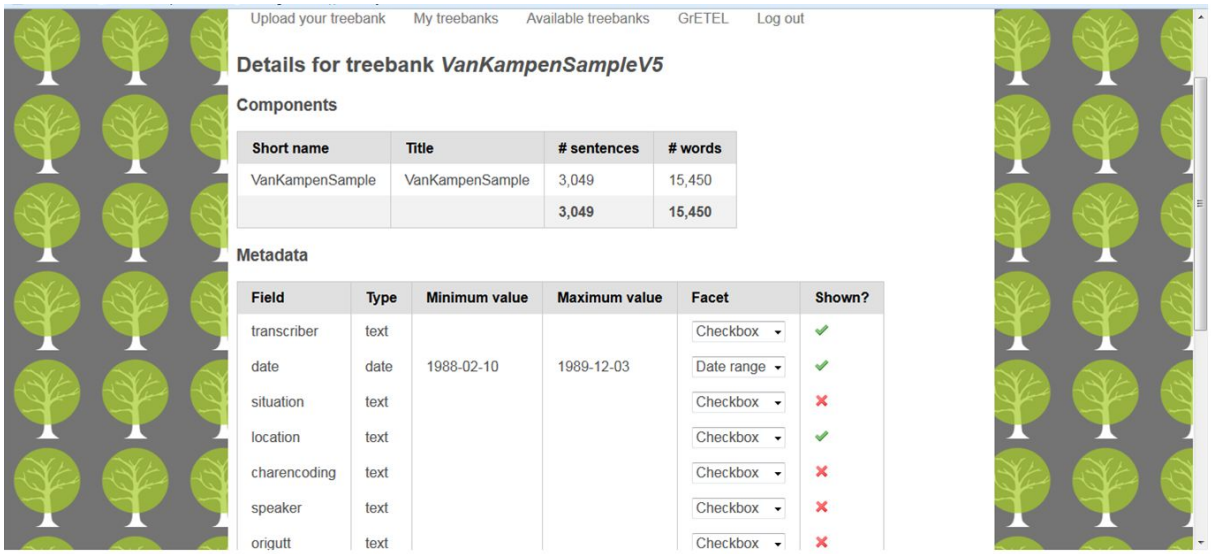


Figure 3: Screenshot of the details page of a particular treebank.

Alpino_ds DTD[10]), or a text corpus. The text files of a text corpus can be in plain text format, or in the CHAT format. In the latter case, the software uses, inter alia, the cleaning algorithm described in section 3.1. We are currently working on providing a wider range of input formats (in particular, FoLiA (van Gompel and Reynaert, 2013) and TEI).[11] Files in plain text or CHAT format are automatically parsed by Alpino. If needed, one can download the automatically parsed corpus, manually correct it or a part of it, and then upload the improved treebank in GrETEL.

Uploading a corpus requires authentication. Currently, this is restricted to users with an Utrecht University account, though a guest account is provided as well. When the extensions are complete, the application (and the treebanks) will be hosted by a certified type B CLARIN centre, most probably the Dutch Language Institute,[12] which will provide CLARIN-compatible federated login.

The maximum size of uploaded corpora will be determined by the CLARIN centre that will host the application. It is likely that a size restriction will be imposed allowing only corpora of maximally a few million words.[13] For larger corpora, it makes more sense to make special arrangements with the CLARIN centre. Very large corpora may require dedicated indexing techniques, e.g. the ones proposed by (Vandeghinste and Augustinus, 2014) and (Vanroy et al., 2017) for dealing with the 510 million word occurrences (41 million utterances) SoNaR corpus.

For representing metadata of corpora, we use a format defined during the development of PaQu that allows users to incorporate metadata in the running text (see `http://www.let.rug.nl/alfa/paqu/info.html#cormeta` for details). Metadata in CHAT files are converted to this format. The software reads in the metadata and will create faceted search in GrETEL to allow users to both analyse and filter their search results.

GrETEL 4 offers new functionality (not present in earlier versions) to further analyse a result set of interest via an analysis interface. This interface enables the creation of pivot tables and graphs such as a heatmap and a table bar map, which allows rapid insight into the data. The result set can also be exported to a tab-separated value text format to allow further analysis in other tools.

The user can not only select metadata elements and their values in this analysis interface but also select words that match with a node in the query tree, as illustrated in section 5.

[10] `http://www.let.rug.nl/vannoord/alp/Alpino/versions/binary/latest.tar.gz`.
[11] `http://www.tei-c.org/`.
[12] `http://ivdnt.org/`.
[13] The XML database BaseX has a theoretical limit of 500GB of XML, according to (Grün, 2010, section 2.4).

5 Example query and analysis

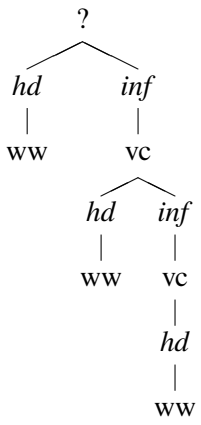
We will illustrate the query and analysis options with an example. We are interested in constructions with three bare[14] verbs in the children's speech. An example sentence illustrating this construction is given in (4), which contains the 3 bare verb forms *zal*, *willen* and *doen*:

(4) Hij zal dat willen doen
He will that want do
'He will want to do that'

In this sentence, the words *hij* 'he' and *dat* 'that' are not essential for the construction that we are interested in, so we mark them as optional. As to the three verbs in this sentence, they are crucial for this construction, but we are not interested in these specific verbs but in any word of category verb that can occur in this construction. Therefore we indicate for these words that we want any word here with the same part of speech. The example sentence is a main clause, but we want to find examples of this construction in any type of clause. Therefore we mark the option 'ignore properties of the dominating node'.

Specifying this results in the XPath query (6), visualised by the query tree (5):

(5)



(6)
```
//node[@cat and
       node[@rel="hd" and @pt="ww"] and
       node[@cat="inf" and @rel="vc" and
            node[@pt="ww" and @rel="hd"] and
            node[@rel="vc" and @cat="inf" and
                 node[@rel="hd" and @pt="ww"]]]]
```

Executing the query on the corpus VKLaura ('Van Kampen Corpus LAURA') yields 325 matches in 325 utterances. A screenshot of the results is shown in Figure 4.

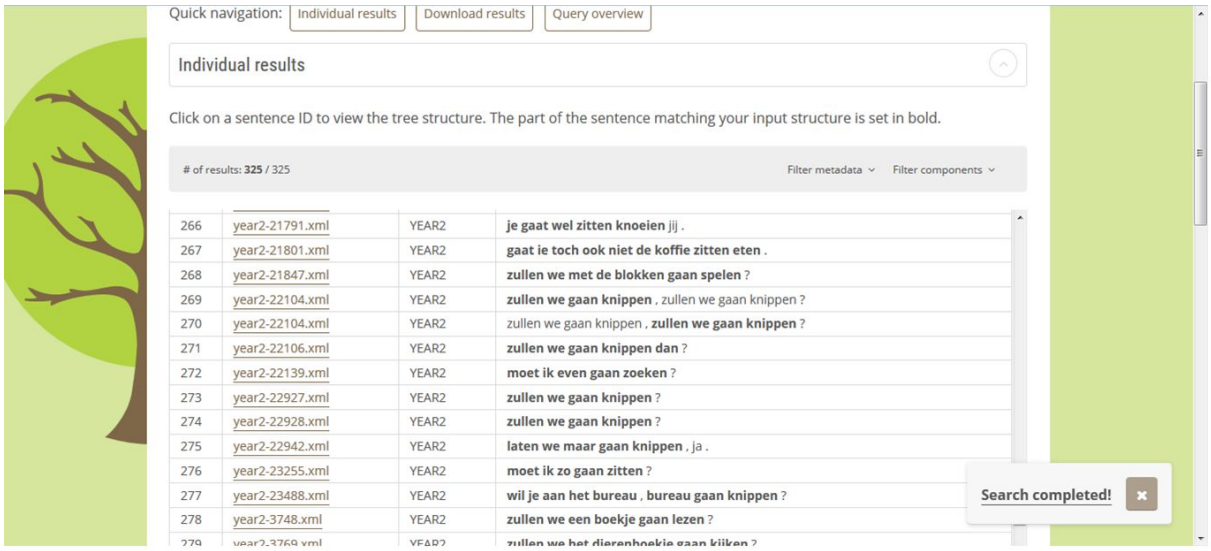


Figure 4: Screenshot of the results page of the query.

[14] i.e. verbs without *te* (cf. English 'to').

We can now filter by metadata and by components. If we filter by *speaker* and select only the (child) speaker LAU (Laura), we obtain 12 matches in 12 utterances (see Figure 5).[15]

Figure 5: Screenshot of the results page of the query after filtering for speaker=LAU.

The search application allows a more detailed analysis of the search results, in particular selecting parts of the result data and metadata, grouping, filtering and sorting them, and represent them in pivot tables or frequency lists, with various visualisation options. For example, we can make a table that shows at what age the speaker LAU has uttered such constructions (as of month 43); or we can create a frequency list of the verb combinations that occur in the results, grouped by speaker (see Figure 6).

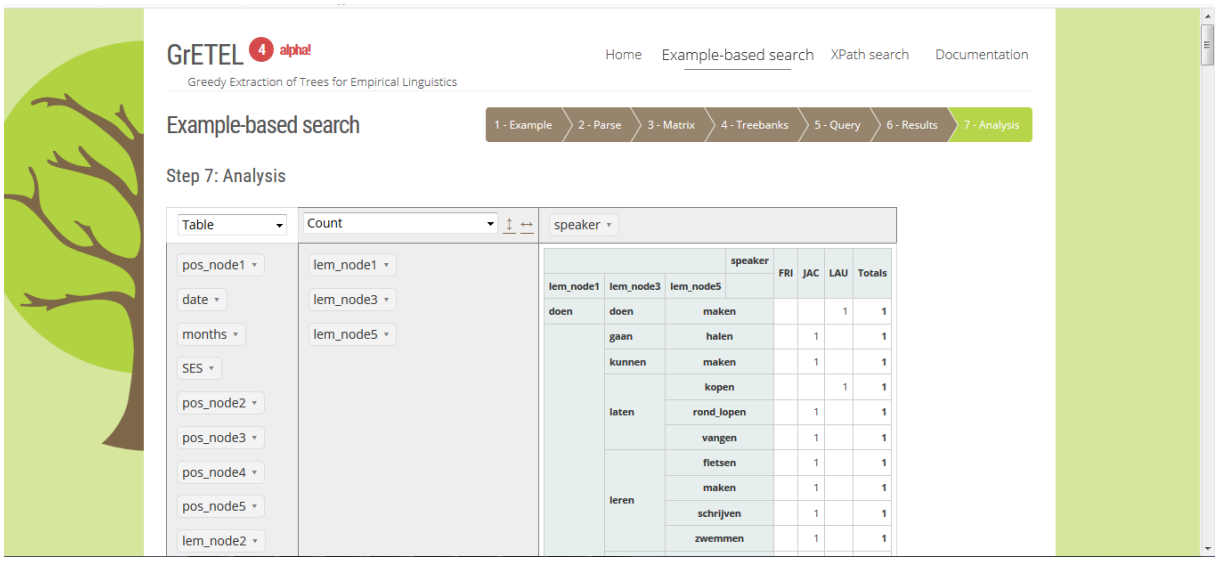


Figure 6: Analysis: verb lemmas used grouped by speaker

These data show that the child uses combinations of three bare verbs, and in only 4 out of the 12 examples the child uses a verb combination that also occurs in the adult's utterances in the corpus. In addition, the ones that the child uses are not from the most frequent verb combinations used by the adult. All of this suggests that the child fully commands the use of such constructions and can creatively use

[15] In the result set three speakers occur, with codes JAC (in the role of mother), LAU (with role target child), and FRI (another adult).

them.[16] Possibly the child has made a generalisation on the basis of the use of constructions with two bare verbs (which occur much more frequently: 6,645 in total, 1,363 uttered by Laura) and are used by the child much earlier (as of month 23).

6 Related Treebank Query Applications

We had two requirements on a treebank query application: (1) it must be compatible with the format generated by the Alpino parser and used in Dutch treebanks such as LASSY (van Noord et al., 2013) and the Spoken Dutch Treebank (Oostdijk et al., 2002)); (2) it must provide a user friendly interface that enables a researcher to query the treebank without having to write a query in a formal query language.

There are several treebank query applications, e.g. PMLTQ (Pajas et al., 2009); the WebLicht application Tündra (Hinrichs et al., 2010), and INESS (Rosén et al., 2012). However, only two treebank query applications meet these requirements: PaQu[17] (Odijk et al., to appear 2017) and GrETEL[18] (Augustinus et al., 2012).

The PaQu (Parse and Query) application enables upload of one's own corpus and provides a user-friendly interface for searching for syntactic dependency relations between words. It also offers facilities for analysis of the query results. PaQu was actually developed on the basis of the LASSY Word Relations application (Tjong Kim Sang et al., 2010) at the request of one of the authors of this paper. In addition, when we started our work on the AnnCor project, the PaQu developers made available a treebank for the Dutch CHILDES corpora with fully automatically parsed utterances in the PaQu application.

Nevertheless, we selected the GrETEL application because it makes it possible to search for arbitrary constructions using example-based querying (e.g. the construction with three verbs can only be queried in PaQu by writing an XPATH query from scratch), and we wanted to offer more sophisticated analysis options than PaQu provides, in particular more sophisticated ways of selecting parts of query results. Furthermore, the analysis interface is more user-friendly by allowing the creation of pivot tables through dragging attributes into a table.

7 Conclusions and Future Work

We have described the extensions to the GrETEL treebank query application we made in the context of the AnnCor project and illustrated it with a query in (a preliminary version of) the AnnCor CHILDES Treebank for Dutch. The extensions involve functionality for uploading one's own corpus with metadata, and functionality for analysing data, subparts of data and metadata in combination. The treebank query application and the treebank are still under development, but the extensions to the GrETEL query application described here are already available.[19] The source code is available on GitHub.[20]

There are a number of aspects of the treebank query application that we would like to work on in the future: (1) extend input formats (FoLiA and TEI); (2) allow more complex metadata that specify properties of spans of text such as retracings, repetitions, pronunciation, paralinguistic material etc. as in example (1);[21] (3) extend the analysis component with frequencies of constructions relative to the size of a subpart of the corpus (e.g. component, session) measured in terms of the number of tokens or number of utterances; and (4) provide a graphical interface for selecting nodes from a query tree in the analysis component.

Acknowledgements

This work was financed by the Utrecht University internal AnnCor research infrastructure project and by the CLARIAH-CORE project funded by the Dutch National Science Foundation (NWO).

[16] Though of course, it is no conclusive evidence, if only because the corpus is just a small sample of the full input of the child and its own production.

[17] http://portal.clarin.nl/node/4182.

[18] http://portal.clarin.nl/node/1967.

[19] via the url http://gretel.hum.uu.nl/gretel4/.

[20] https://github.com/UUDigitalHumanitieslab/gretel, https://github.com/UUDigitalHumanitieslab/GrETEL-upload.

[21] See (MacWhinney, 2015) for many more examples.

References

Liesbeth Augustinus, Vincent Vandeghinste, and Frank Van Eynde. 2012. Example-based treebank querying. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), Istanbul, Turkey.

Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2001. Alpino: Wide-coverage computational analysis of Dutch. *Language and Computers* 37(1):45–59.

C. Grün. 2010. *Storing and querying large XML instances*. Ph.D. thesis, University of Konstanz, Konstanz, Germany. http://nbn-resolving.de/urn:nbn:de:bsz:352-opus-127142.

Erhard W. Hinrichs, Marie Hinrichs, and Thomas Zastrow. 2010. WebLicht: Web-Based LRT Services for German. In *Proceedings of the ACL 2010 System Demonstrations*. pages 25–29. http://www.aclweb.org/anthology/P10-4005.

Brian MacWhinney. 2000. *The CHILDES Project: Tools for Analyzing Talk*. Lawrence Erlbaum Associates, Mahwah, NJ, 3rd edition.

Brian MacWhinney. 2015. Tools for analyzing talk, electronic edition, part 1: The CHAT transcription format. Technical report, Carnegie Mellon University, Pittsburg, PA. http://childes.psy.cmu.edu/manuals/CHAT.pdf.

Jan Odijk. 2015. Linguistic research with PaQu. *Computational Linguistics in the Netherlands Journal* 5:3–14. http://www.clinjournal.org/sites/clinjournal.org/files/odijk2015.pdf.

Jan Odijk. 2016a. A Use case for Linguistic Research on Dutch with CLARIN. In Koenraad De Smedt, editor, *Selected Papers from the CLARIN Annual Conference 2015, October 14-16, 2015, Wroclaw, Poland*. CLARIN, Linköping University Electronic Press, Linköping, Sweden, number 123 in Linköping Electronic Conference Proceedings, pages 45–61. http://www.ep.liu.se/ecp/article.asp?issue=123&article=004, http://dspace.library.uu.nl/handle/1874/339492.

Jan Odijk. 2016b. Linguistic research using CLARIN. *Lingua* 178:1 – 4. Linguistic Research in the CLARIN Infrastructure. https://doi.org/http://dx.doi.org/10.1016/j.lingua.2016.04.003.

Jan Odijk, Alexis Dimitriadis, Martijn van der Klis, Marjo van Koppen, Meie Otten, and Remco van der Veen. 2017. The AnnCor CHILDES Treebank. Unpublished paper, AnnCor project, Utrecht University. accepted for LREC 2018.

Jan Odijk and Arjan van Hessen, editors. to appear 2017. *CLARIN in the Low Countries*. Ubiquity Press, London, UK. To appear as Open Access.

Jan Odijk, Gertjan van Noord, Peter Kleiweg, and Erik Tjong Kim Sang. to appear 2017. The parse and query (PaQu) application. In Jan Odijk and Arjan van Hessen, editors, *CLARIN in the Low Countries*, Ubiquity, London, UK, chapter 23. DOI: http://dx.doi.org/10.5334/bbi.23. License: CC-BY 4.0.

N. Oostdijk, W. Goedertier, F. Van Eynde, L. Boves, J.P. Martens, M. Moortgat, and H. Baayen. 2002. Experiences from the Spoken Dutch Corpus project. In M. González Rodriguez and C. Paz Suárez Araujo, editors, *Proceedings of the third International Conference on Language Resources and Evaluation (LREC-2002)*, ELRA, Las Palmas, pages 340–347.

Petr Pajas, Jan Štěpánek, and Michal Sedlák. 2009. PML tree query. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, Prague. http://hdl.handle.net/11858/00-097C-0000-0022-C7F6-3.

Victoria Rosén, Koenraad De Smedt, Paul Meurer, and Helge Dyvik. 2012. An open infrastructure for advanced treebanking. In Jan Hajič, Koenraad De Smedt, Marko Tadić, and António Branco, editors, *META-RESEARCH Workshop on Advanced Treebanking at LREC2012*. ELRA, Istanbul, Turkey, pages 22–29.

Kenji Sagae, Eric Davis, Alon Lavie, Brian MacWhinney, and Shuly Wintner. 2007. High-accuracy annotation and parsing of CHILDES transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*. Association for Computational Linguistics, Stroudsburg, PA, USA, CACLA '07, pages 25–32. http://dl.acm.org/citation.cfm?id=1629795.1629799.

Erik Tjong Kim Sang, Gosse Bouma, and Gertjan van Noord. 2010. LASSY for beginners. Presentation at CLIN 2010, Utrecht. http://ifarm.nl/erikt/talks/clin2010.pdf.

Maarten van Gompel and Martin Reynaert. 2013. FoLiA: A practical XML format for linguistic annotation - a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal* 3:63–81.

Gertjan van Noord, Gosse Bouma, Frank Van Eynde, Daniël de Kok, Jelmer van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2013. Large scale syntactic annotation of written Dutch: Lassy. In Peter Spyns and Jan Odijk, editors, *Essential Speech and Language Technology for Dutch*, Springer Berlin Heidelberg, Theory and Applications of Natural Language Processing, pages 147–164. https://doi.org/10.1007/978-3-642-30910-6$_9$.

Vincent Vandeghinste and Liesbeth Augustinus. 2014. Making large treebanks searchable. the SONAR case. In *Proceedings of the LREC 2014 2nd workshop on Challenges in the Management of Large Corpora (CMLC-2)*. Reykjavik, pages 15–20. http://www.lrec-conf.org/proceedings/lrec2014/workshops/LREC2014Workshop-CMLC2%20Proceedings-rev2.pdf.

Bram Vanroy, Vincent Vandeghinste, and Liesbeth Augustinus. 2017. Querying large treebanks: Benchmarking GrETEL indexing. *Computational Linguistics in the Netherlands Journal* 7:145–166.